

## **NoobaVSS: Video Processing Framework to Enhance Processing and Automated Manipulation of Surveillance Videos**

A. Nanayakkara, A. Dissanayaka, C. Wijenayake, C. Hettiarachchi  
*Department of Computer Science & Engineering, University of Moratuwa, Katubedda, Sri Lanka*

*and*

C. De Silva  
*Senior Lecturer, Department of Computer Science & Engineering, University of Moratuwa, Katubedda, Sri Lanka*

### **Introduction**

Surveillance cameras are becoming artificial eyes capable of monitoring behaviors, activities, or other visual information with the purpose of influencing, managing, directing, or protecting. However they still depend on human assistance in interpreting any anomalies in the scenes they capture. Next generation smart surveillance systems are expected to be capable of detecting anomalies by themselves releasing human operators from constant, manual observation of the video feeds. In the recent past Sri Lanka has shown a rapid increase in the use of CCTV surveillance systems in different types of environments including commercial, non-commercial and government sectors. Most of these however are used only for post-incident investigation purposes mainly due to the higher effort and cost required for real time analysis. The unavailability of video analyzing platforms in the public domain and non-existence of open source video analyzing software has deterred their use for pre-incident investigation and real time analysis. Our research effort is to develop a software framework that will act as a testing framework and software basement for automated surveillance video analysis with the aim of improving quality and level of security provided by video surveillance systems. A sample scenario for a banking environment is studied extensively to guide the development process.

### **Methodology**

The framework is developed as a component based model. A set of individual plugins have been developed separately and connected to the main engine where each individual plugin is responsible for a separate feature extracting task. A plug-in is basically capable of processing a given sequence of image frames from a video and extract designated features (ex: Number of faces in the scene, Speed of an object in the scene). To identify these key features to be extracted from the video imagery, a scenario analysis is conducted over capturing domain (in our extensive study-banking environment). Scenario analysis is useful in identifying what is needed to be extracted from the input video and what is not needed to be extracted. Since the approach in writing scenarios is not restricted to any formal method or constrained by any event sequence, more free flowing and different scenarios are captured. These scenarios ultimately make it easier to identify the nature of the environment and give more insight in identifying computer vision techniques that need to be used.

Next, to extract each of those features of the video, a separate plugin has been developed. Knowledge representation platform has been developed using the Qt framework. This framework has the unique capability of loosely coupling functions using signal slot mechanism. Each processing plugin essentially has the same structure, where it may or may not subscribe to outputs of some other plugins. It processes the inputs accordingly within the given time frame and emits its output, if any. They all are feature detectors which take input from a surveillance video feed. A global timing signal has been used to keep track of time and an abstract processing node facilitates signal slot mechanism. It has an abstract process method, so that the processing modules inherited from it can implement a different functionality for a process

method. However, nodes named as D does not subscribe to any other nodes. They can be feature detectors which take input from a video feed. In the testing environment, it can read from a file and emit the content as an event for the given time frame.

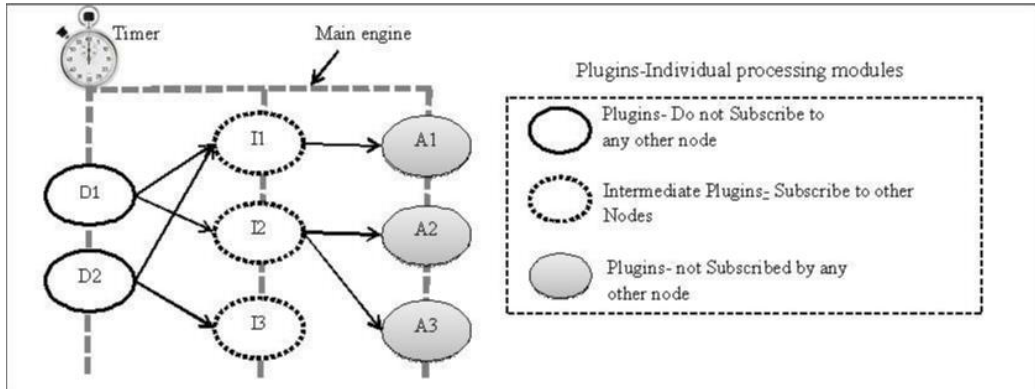


Figure 1. High-level System Architecture of the NoobaVSS processing platform.

## Results and Discussion

The proposed solution is currently capable of detecting faces, tracking, detecting moving speed, object relative distance calculation, loitering detection and abandoned object detection. Processing can be deployed for real-time as well as offline video. The plugin based model enables any other video processing or feature detection component to be added to the framework, testing algorithms and further extending them for smart video surveillance enhancements. In the extensive study, we have been able to apply NoobaVSS to demonstrate a detection of a bag abandoning situation. The framework can also be extended to fit for computer vision domains other than surveillance and security. Since, no other proper open source video processing and testing platform is currently available in the domain this framework can emerge as promising tool for computer vision based testing platform for surveillance. As the processing framework is cross platform and open source, it can be deployed platform independently.

## Conclusions

Sri Lanka is emerging as a significant contributor to the open source community and has the potential of improving economical and intellectual values of the country via introducing innovative software solutions. This research project can be identified as a basement for several image and video analysis solutions including automation of suspicion event detection of surveillance video, hence improve security and safety. For an example use case, this can lead a path towards developing smart surveillance systems for critical environments such as banking environments, railway stations and shopping malls with the capability of automated identification of suspicious events and alert generation upon such circumstances.

## References

- Krausz, B., Herpers, R., 2008. Event detection for video surveillance using an expert system in AREA.
- Viola, P., Jones, M., 2004. Robust real-time face detection. International Journal of Computer Vision (IJCV). 137-154.
- Sobral, A., 2013. BGSLibrary: An OpenCV C++ Background Subtraction Library". IX Workshop de Visão Computacional.